



# **Transformaciones Modelo a Modelo y Modelo a Código con IBM Rational Software Architect**

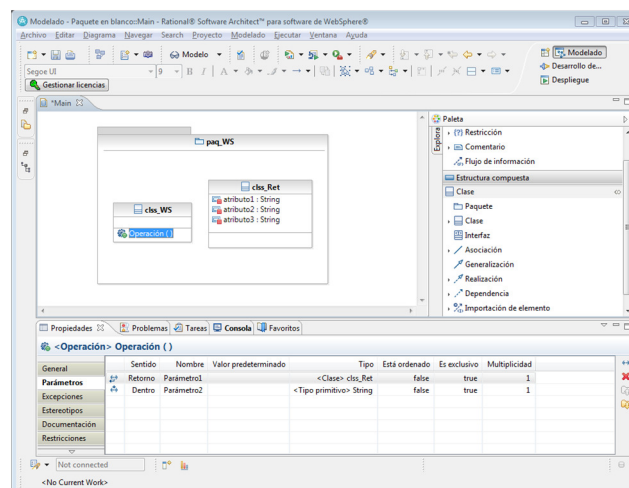
Alfonso García de Prado Fontela  
Ingeniería en Automática, Electrónica, Arquitectura y Redes de  
Computadores  
Universidad de Cádiz

## Transformaciones Modelo a Modelo y Modelo a Código con IBM Rational Software Architect

*IBM Rational Software Architect* (RSA) es un entorno optimizado -tanto para Windows como para Linux- para el de modelado y desarrollo de aplicaciones C++ y JavaEE. Este entorno está basado en Eclipse y está preparado para poder desarrollar aplicaciones siguiendo los principios del desarrollo dirigido por modelos: realizar transformaciones entre modelos, de modelo a código, de código a modelo. También está preparado para el desarrollo de servicios Web de forma convencional o a partir de modelos UML. Al estar basado en eclipse es compatible con la mayoría de los *plugins* que este soporta, como el que da soporte a AspectJ (AJDT). Un inconveniente es que no es software libre, aunque se puede obtener una versión de prueba de 30 días en la página oficial<sup>1</sup>; además IBM tiene una política abierta de licencias en sus productos de desarrollo para el entorno universitario e investigación. Para beneficiarse de esta política solo hay que registrarse en su página oficial<sup>2</sup>.

### Modelar y generar transformaciones modelo a modelo

Para crear un proyecto de modelado basta con hacer *click* en *File/new/proyect/UML Project* y seguir el *wizard*, el cual nos permitirá especificar el tipo de modelo que vamos a realizar, en nuestro caso un diagrama de clases. Una vez realizado esto podremos dibujar el diagrama de clases con los componentes que tenemos en la parte derecha de la ventana y establecer todas las propiedades en la zona inferior. Podemos ver un ejemplo en la Figura 1.



**Figura 1** – Creación de un modelo UML en RSA

RSA nos proporciona una serie de transformaciones comunes, aunque lo habitual es que nuestras necesidades de diseño nos obliguen a generar nuestras propias transformaciones. En las siguiente líneas explicaremos los pasos para crear una transformación modelo a modelo.

Para empezar hay que crear un proyecto de transformación de correlación modelo a modelo haciendo *click* en *File/New/Project/Model to Model Mapping Transformation Project*. En la tercera pantalla del *wizard*, donde hay que seleccionar la plantilla, habrá que elegir *Plug-in with Transformation Mapping*. En la siguiente página del *wizard* tendremos que seleccionar los metamodelos de origen y destino, en nuestro caso el metamodelo de UML2. En la Figura 2 podemos encontrar la secuencia para encontrarlo.

<sup>1</sup> <http://www14.software.ibm.com/webapp/download/search.jsp?pn=Rational+Software+Architect>

<sup>2</sup> <http://www.ibm.com/developerworks/university/membership/index.html>

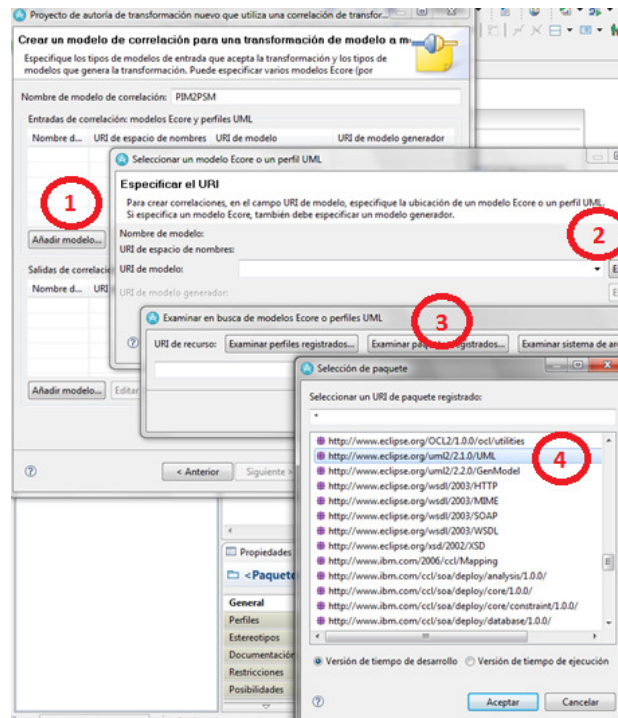


Figura 2 – Pasos para encontrar el metamodelo de UML

Una vez tenemos el proyecto creado tenemos que hacer las correlaciones entre los elementos del origen y el destino: para crear una nueva correlación hay que hacer *click* con el botón derecho en la correlación raíz y seleccionar *new map* (Figura 3, marca 1), darle el nombre y seleccionar con los botones que indicamos en la marca 2 el tipo de elementos de origen y de destino de la correlación. En este punto solo hay que establecer las equivalencias y transformaciones entre origen y destino (marca 3). Si necesitamos establecer algo más específico, podemos personalizar estas transformaciones usando código como vemos en la Figura (marca 4).

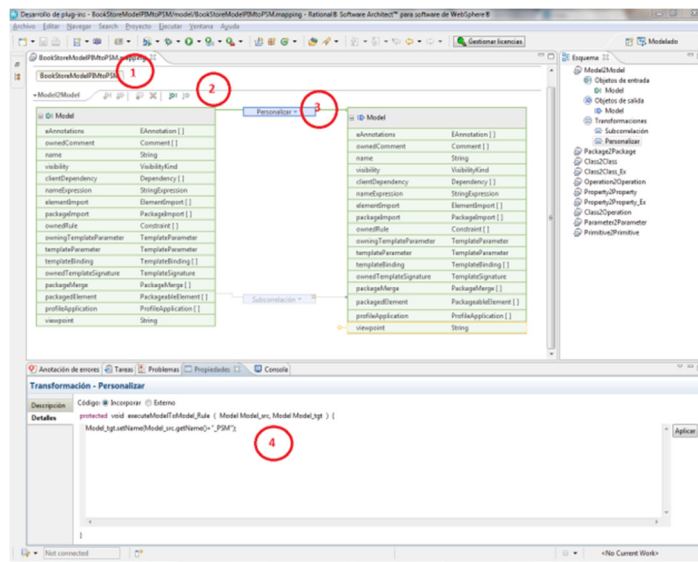


Figura 3 – Pasos para crear las correlaciones

Una vez realizadas todas las correlaciones de nuestra transformación, y para realizar nuestra transformación modelo a modelo, tenemos que generar el código de la correlación, este es un proceso automático que se inicia al pulsar con el botón derecho en nuestro fichero de correlación con extensión *“.mapping”* y seleccionando *Transform/Generate Transformation Code*. Generado el código hay que ejecutar una nueva instancia del RSA que contenga el *plugin* que acabamos de generar y que nos permitirá realizar la transformación modelo a modelo. Para ello tenemos que pulsar de nuevo con el botón derecho sobre nuestro

fichero de correlación y seleccionar *Run as/Run Configurations* y seguir los pasos indicados en la Figura 4.

La nueva instancia de RSA, al ser para probar el *plugin*, nos abre el RSA en un nuevo directorio de trabajo que en un principio estará vacío. Si aún no hemos creado el modelo lo podemos crear aquí, si ya lo tenemos creado lo podemos importar del otro área de trabajo, o bien podemos generar el *plugin* para que se abra con la instancia primaria de RSA. Suponiendo que ya está creado el proyecto UML con el modelo PIM, para transformar el modelo base al nuevo modelo usando las reglas de transformación que hemos diseñado, tenemos que hacer *click* en *File/New/Transformation Configuration*, seleccionar la transformación que hemos generado y seguir los pasos restantes del wizard.

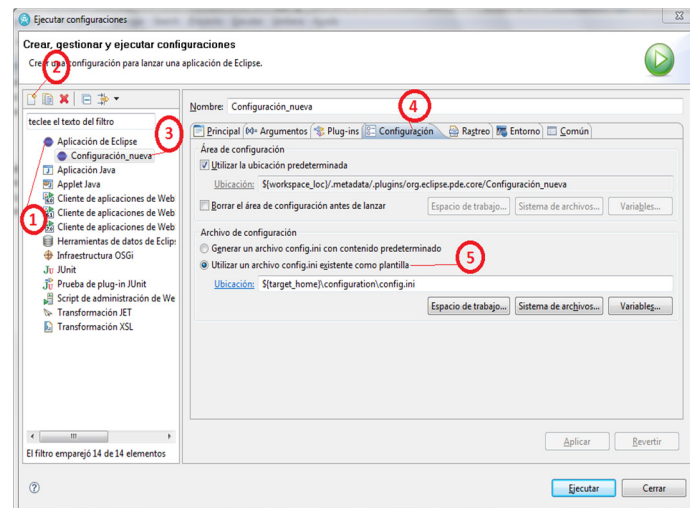


Figura 4 – Ejecución de una instancia de RSA con el *plugin* generado

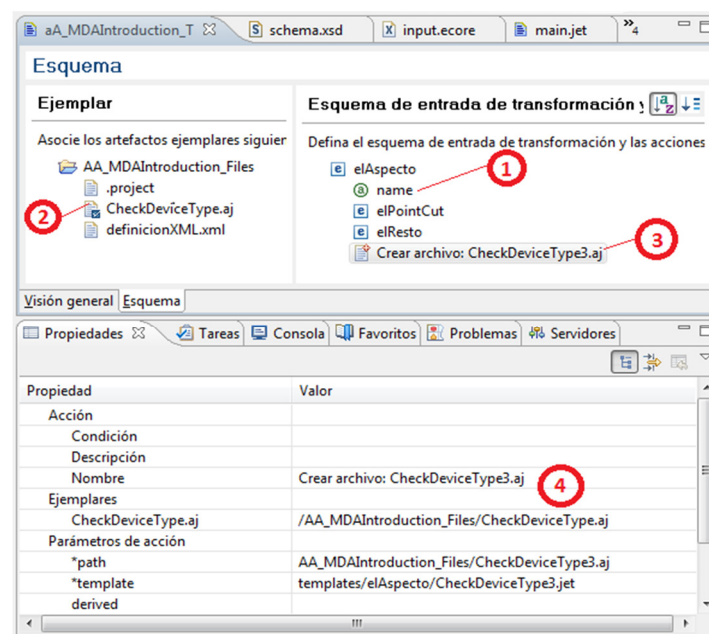


Figura 5 – Creación de un esquema de transformación

## Generar transformaciones modelo a código usando plantillas

Una de las formas para generar código es partiendo de plantillas o ficheros de ejemplo. Para este propósito usaremos el *plugin* de Eclipse *JET with exemplars*, este *plugin* lo tiene RSA en su paquete de instalación.

Para crear la transformación con este *plugin* solo hay que seguir los siguientes pasos:

- 1) Crear un proyecto general que contendrá la plantilla origen en la que se basará para realizar la transformación.
- 2) Crear un “proyecto con autoría de ejemplar” haciendo *click* en *File/New/Project/EMFT JET Project with Exemplar Authoring*. En la tercera pestaña del *plugin* hay que seleccionar el proyecto que hemos creado en el paso 1, el que tiene la plantilla origen.
- 3) Realizar el esquema de transformación, añadiendo los tipos y atributos que definen nuestro modelo UML de entrada (Figura 5, paso 1).
- 4) Desplazar (*Drag&Drop*) el fichero plantilla que está en el lado izquierdo de la ventana de esquema al lado derecho (pasos 2 y 3). Hay que seleccionar el nuevo elemento que se ha creado al desplazar el fichero, y cambiarle el nombre y la ruta para que no sobrescriba el original al realizar la transformación (paso 3). Hacer *click* con el botón derecho sobre el elemento *root* del esquema y pulsar en actualizar.
- 5) Abrir el fichero “*.jet*” en el directorio y remplazar en él las partes que van a ser actualizadas con los datos de entrada.
- 6) Crear y hacer *click* con el botón derecho en el fichero fuente y ejecutar como *run configuration/ Input for jet transformation*.